CMPSC 465 Data Structures & Algorithms
Spring 2024 Mehrdad Mahdavi and David Koslicki

HW 8

Due March 28, 10:00 pm

Instructions: You are encouraged to solve the problem sets on your own, or in groups of up to five people, but you must write your solutions strictly by yourself. You must explicitly acknowledge in your write-up all your collaborators, as well as any books, papers, web pages, etc. you got ideas from.

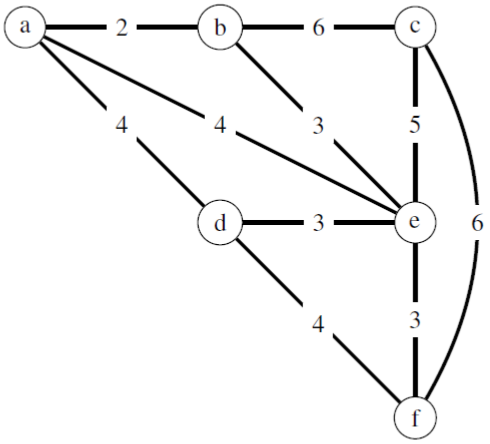
Formatting: Each problem should begin on a new page. Each page should be clearly labeled with the problem number. The pages of your homework submissions must be in order. You risk receiving no credit for it if you do not adhere to these guidelines.

Late homework will not be accepted. Please, do not ask for extensions since we will provide solutions shortly after the due date. Remember that we will drop your lowest two scores.

You need to submit it via Gradescope (Class code XX7RVV). Please ask on Campuswire about any details concerning Gradescope and formatting.

For each algorithm question, explain your algorithm and analyze its correctness and running time. Pseudocode is not required, but you may include it if you feel it makes your written explanation more clear.

1. (20 pts.) **Kruskal's Algorithm.** Run Kruskal's algorithm on the following undirected graph. Give the order of edges that are added to the minimum spanning tree (whenever you have a choice, always choose the smallest edge in lexicographic order). Whenever an edge is added, show the current vertex set(s).



2. (20 pts.) **Disjoint Set Operations.** You are given seven items labeled $0, 1, \ldots, 6$, each of which initially belongs to its own set: $\{0\}, \{1\}, \ldots, \{6\}$. For parts (a) and (b), start from these initial sets and show the state after each `union` operation by drawing a forest. Each set should be represented as a tree, with seven single-node trees before the first `union` and a single seven-node tree after the final `union`. The operations are as follows: `union`$(0,3)$, `union`$(3,4)$, `union`$(2,5)$, `union`$(6,1)$, `union`$(0,6)$, `union`$(5,1)$.

(a) Show the results of the above `union` operations. Given `union`(x,y), always choose to merge x's tree into y's tree.

(b) Show the results of the above `union` operations, always choosing to merge the tree with fewer elements into the tree with more elements (breaking ties by using the approach from part (a)).

(c) Compare the number of nodes that must be visited to compute `root`(0) in the final trees for parts (a) and (b).

(d) Now consider the worst-case series of `union` operations for each method (a) and (b), and the final trees created by these operations. Compare the maximum number of nodes that must be visited when performing a `root` operation on these two worst-case trees.

3. (20 pts.) **Maximum spanning tree.**

Design two approaches for the maximum spanning tree problem, i.e., given an undirected graph $G = (V, E)$ with edge weight $w(e)$ for any $e \in E$, to compute a spanning tree T of G such that $\sum_{e \in T} w(e)$ is maximized. They need to have the same running time as Kruskal's algorithm.

4. (20 pts.) **MST Basics.**

For each of the following statements, either prove or supply a counterexample. Always assume $G = (V, E)$ is undirected and connected. Do not assume the edge weights are distinct unless specifically stated.

1. Let e be any edge of minimum weight in G. Then e must be part of some MST.
2. If e is part of some MST of G, then it must be a lightest edge across some cut of G.
3. If G has a cycle with a unique lightest edge e, then e must be part of every MST.
4. For any $r > 0$, define an r-path to be a path whose edges all have weight less than r. If G contains an r-path p from s to t, then every MST of G must also contain an r-path (not necessarily identical to p) from s to t.

5. (20 pts.) **Critical edge.** An edge of an $s-t$ flow network is called *critical* if decreasing the capacity of this edge results in a decrease in the $s-t$ maximum flow. Give an efficient algorithm that finds a critical edge in a network.

6. (0 pts.) **Acknowledgments.** The assignment will receive a 0 if this question is not answered.
(a) If you worked in a group, list the members of the group. Otherwise, write "I did not work in a group."
(b) If you received significant ideas about the HW solutions from anyone not in your group, list their names here. Otherwise, write "I did not consult with anyone other than my group members."
(c) List any resources besides the course material that you consulted in order to solve the material. If you did not consult anything, write "I did not consult any non-class materials."